

1) $Z\text{-number} = -1$

Áp dụng công thức $Z = \frac{x - \mu}{\sigma}$

$$\frac{-1 - 3}{2} = -2 = Z$$

tiếp tục với $Z\text{-number} = 3$

$$\Rightarrow Z = \frac{3 - 3}{2} = 0$$

$Z\text{-number} = 9$

$$\Rightarrow Z = \frac{9 - 3}{2} = 3$$

$$\Rightarrow Z = \{-2, 0, 3\}$$

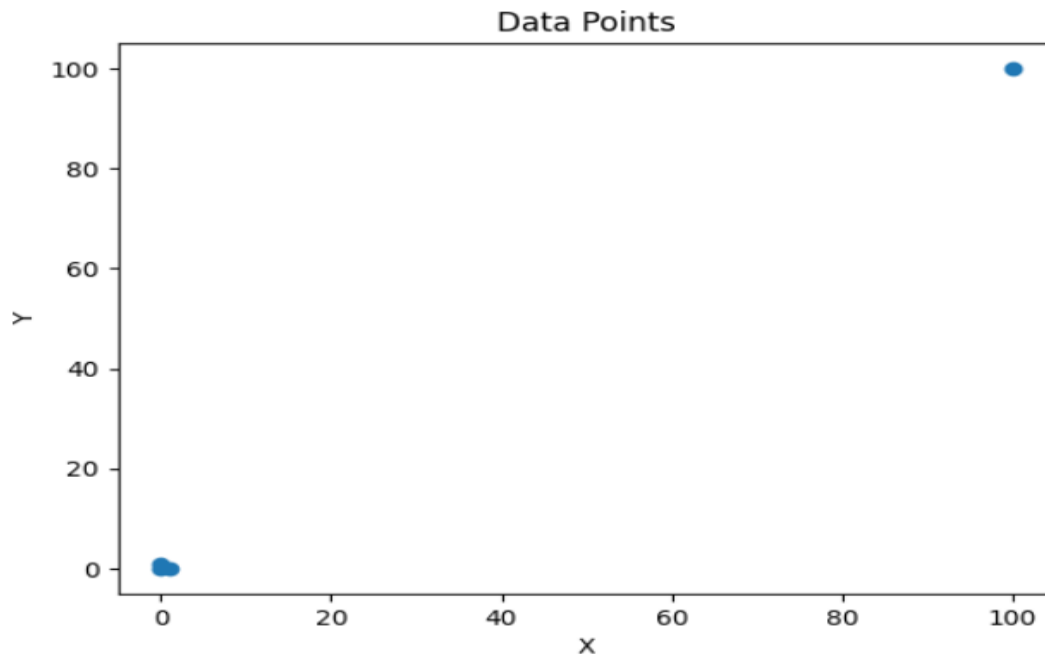
$\max|Z| = 3 \Rightarrow$ vậy 9 được xem là giới hạn lớn nhất trong trường hợp này

2) ta có công thức

$$d(P, Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \dots + (p_n - q_n)^2}$$

```
import numpy as np
import matplotlib.pyplot as plt

data = np.array([[0, 0], [0, 1], [1, 0], [100, 100]])
plt.scatter(data[:, 0], data[:, 1])
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Data Points')
plt.show()
```



Vẽ 4 điểm

```
def distance(point, mean, cov):
    inv_cov = np.linalg.inv(cov)
    diff = point - mean
    mahalanobis_dist = np.sqrt(np.dot(np.dot(diff, inv_cov), diff.T))
    return mahalanobis_dist
```

Tính khoảng cách mahalanobis

```

def find(data):
    # mean và cov
    mean = np.mean(data, axis=0)
    cov = np.cov(data.T)

    # Tính toán giá trị Mahalanobis cho mỗi điểm dữ liệu
    m_values = []
    for point in data:
        m_dist = distance(point, mean, cov)
        m_values.append(m_dist)

    # Tìm điểm cực đại dựa trên Mahalanobis
    max_index = np.argmax(m_values)
    mahalanobis = data[max_index]

    # In kết quả
    print("cực trị : (100, 100)")
    print("(Mahalanobis):", mahalanobis)

    # Tìm điểm cực trị độ sâu lớn nhất
    depth_values = np.array(m_values) ** 2
    max_depth_index = np.argmax(depth_values)
    max_depth_point = data[max_depth_index]

    # In kết quả
    print(" độ sâu :", max_depth_point)

```

Tìm kiếm các trong số

